An Empirical Investigation of Visual Reinforcement Learning for 3D Continuous Control

Samyeul Noh*
ETRI
Daejeon, South Korea
samuel@etri.re.kr

Seonghyun Kim
ETRI
Daejeon, South Korea
kim-sh@etri.re.kr

Ingook Jang
ETRI
Daejeon, South Korea
ingook@etri.re.kr

Donghun Lee
ETRI
Daejeon, South Korea
donghun@etri.re.kr

***Abstract*—Sample-efficient reinforcement learning (RL) methods that can learn directly from raw sensory data will open up real-world applications in robotics and control. Recent breakthroughs in visual RL have shown that incorporating a latent representation alongside traditional RL techniques bridges the gap between state-based and image-based training paradigms. In this paper, we conduct an empirical investigation of visual RL, which can be trained end-to-end directly from image pixels, to address 3D continuous control problems. To this end, we evaluate three recent visual RL algorithms (CURL, SAC+AE, and DrQ-v2) with respect to sample efficiency and task performance on two 3D locomotion tasks ('quadruped-walk' and 'quadruped-run') from the DeepMind control suite. We find that using data augmentation, rather than using contrastive learning or an auto-encoder, plays an important role in improving sample efficiency and task performance in image-based training.**

***Index Terms*—end-to-end reinforcement learning, image-based training, visual reinforcement learning**

I. Introduction

Deep reinforcement learning (deep RL) has proven to be an effective combination of RL with deep learning, enabling remarkable achievements across various domains such as robotics and control [1]–[3]. Many deep RL studies still favor state-based training. This preference arises from the conventional acceptance that learning from coordinate states provides significantly better performance with respect to sample efficiency compared to learning from image pixels. However, relying on state-based training comes with a major limitation: when applying the learned policy to a real-world environment, an additional perception module becomes necessary to acquire the environment state [4]–[8].

Over the past five years, the RL community has made significant progress in improving sample efficiency in image-based training [9]–[11]. These advances in visual RL have been achieved by learning better low-dimensional latent representations through contrastive learning [9], an auto-encoder [10], or data augmentation [11]. In particular, some recent visual RL studies, such as DrQ-v2 based on model-free RL [11] and Dreamer-v2 based on model-based RL [12], have demonstrated that they can successfully solve complex 3D continuous control problems, such as humanoid locomotion tasks from the DeepMind control (DMC) suite [13].

In this paper, we conduct an empirical investigation of visual RL, which can be trained end-to-end directly from image pixels, to address 3D continuous control problems. To this end, we evaluate three recent model-free visual RL algorithms (CURL [9], SAC+AE [10], and DrQ-v2 [11]) on two 3D locomotion tasks ('quadruped-walk' and 'quadruped-run') from the DMC suite. Our experimental results show that CURL based on contrastive learning and SAC+AE based on an auto-encoder struggle to solve the two 3D locomotion tasks while DrQ-v2 based on data augmentation performs well on the tasks. These results indicate that using data augmentation, rather than using contrastive learning or an auto-encoder, plays a crucial role in improving sample efficiency and task performance in image-based training.

The rest of this paper is organized as follows. Section II provides a brief background for visual RL. Section III presents an overview of three recent visual RL algorithms: CURL, SAC+AE, and DrQ-v2. Section IV provides our empirical evaluation of the three visual RL algorithms with respect to sample efficiency and task performance on two 3D locomotion tasks ('quadruped-walk' and 'quadruped-run') from the DMC suite. Finally, Section V provides a summary of the paper.

II. Background

A. Visual Reinforcement Learning

We formulate visual RL as an infinite-horizon partially observable Markov decision process (POMDP) on the basis of RGB images. Such POMDP can be described as a tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma, p_o)$, where $\mathcal{S}$ denotes the state space (a stack of three consecutive RGB images), $\mathcal{A}$ denotes the action space, P is the transition probability, R is the reward function, γ is a discount factor, and p_0 is the distribution of the initial state s_0. The goal is to find a policy π that maximizes the expected discounted sum of rewards $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right]$, where $s_0 \sim p_0$, $a_t \sim \pi(\cdot \mid s_t)$, $s_{t+1} \sim p(\cdot \mid s_t, a_t)$, and $r_t = R(s_t, a_t)$.

B. Soft Actor–Critic

Soft actor–critic (SAC) [14] is an off-policy RL algorithm for continuous control problems that concurrently learns a Q-function, Q_ϕ, a stochastic policy, π_θ, and a temperature, α, on the basis of the maximum entropy framework. With the goal of striking a balance between expected return and entropy, SAC performs soft policy evaluation and improvement. The γ-discounted maximum-entropy objective is $\mathbb{E}_{(s_t,a_t)\sim\mathcal{D}}\left[R(s_t, a_t) + \alpha H(\pi(\cdot \mid s_t))\right]$.

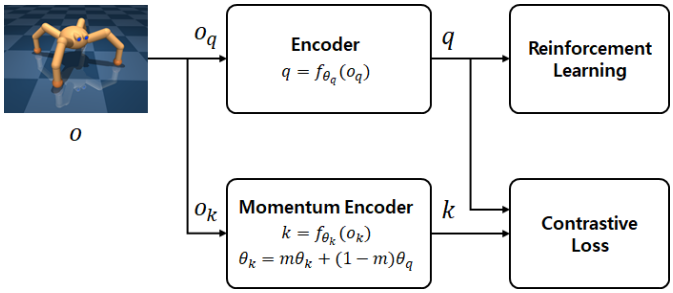


Fig. 1: CURL, which combines instance contrastive learning and RL, learns an encoder by aligning the embeddings of two data-augmented versions, o_q and o_k, derived from an observation, o, using a contrastive loss.

C. Deep Deterministic Policy Gradient

Deep deterministic policy gradient (DDPG) [15] is an off-policy RL algorithm for continuous control problems that concurrently learns a Q-function, Q_ϕ, and a deterministic policy, π_θ. DDPG uses Q-learning to learn Q_ϕ by minimizing one-step Bellman residual $\mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{D}}\left[Q_\phi(s_t,a_t) - r_t - \gamma Q_{\bar{\phi}}(s_{t+1},\pi_\theta(s_{t+1}))\right]$, where $\bar{\phi}$ is an exponential moving average of the weights. The policy π_θ is learned by using Deterministic Policy Gradient (DPG) [16] and maximizing $\mathbb{E}_{s_t\sim\mathcal{D}}\left[Q_\phi(s_t,\pi_\theta(s_t))\right]$.

III. Visual Reinforcement Learning

A. CURL

Contrastive Unsupervised Representations for Reinforcement Learning (CURL) [9] combines two learning techniques, instance contrastive learning and RL. The primary goal of CURL is to train a visual representation encoder, and it achieves this by aligning the embeddings of two data-augmented versions, denoted as o_q and o_k, derived from an image observation o, using a contrastive loss, as shown in Figure 1. The query observations, represented by o_q, act as the reference point, while the key observations, represented by o_k, consist of positive and negative examples generated from the mini-batch employed for the RL update. The key observations are encoded using a momentum-averaged version of the query encoder. CURL integrates with the RL pipeline by constructing the RL policy and/or value function based on the query encoder. Subsequently, these components undergo joint training, incorporating both the contrastive learning and RL objectives. A notable strength of CURL lies in its versatility, as it seamlessly integrates with various RL algorithms that necessitate learning representations from high-dimensional image data.

B. SAC+AE

SAC with Auto-Encoder (SAC+AE) [10] enhances the SAC method by incorporating a regularized auto-encoder, allowing for stable end-to-end training using image pixels in the off-policy regime. The primary goal of SAC+AE is to improve sample efficiency in image-based training, and it achieves this through two key strategies: i) utilizing off-policy methods and ii) employing self-supervised auxiliary losses. In this setting, SAC+AE adopts an auxiliary loss that doesn't rely on task-specific inductive biases, making the approach more robust. Through a comprehensive investigation of combining reconstruction loss with off-policy methods to enhance sample efficiency in settings with rich observations, SAC+AE arrives at two main findings. Firstly, deterministic auto-encoder models perform better than β-VAEs [17] due to additional instabilities stemming from bootstrapping, off-policy data, and joint training with auxiliary losses. Secondly, propagating the actor's gradients through the convolutional encoder negatively impacts performance. The loss function of SAC+AE is as follows:

$$J(RAE) = \mathbb{E}_{o_t\sim\mathcal{D}}\left[\log p_\theta(o_t|z_t) + \lambda_z||z_t||^2 + \lambda_\theta||\theta||^2\right] \quad (1)$$

where $z_t = g_\phi(o_t)$, and λ_z, λ_θ are hyper-paramers.

C. DrQ-v2

DrQ-v2 [11] is a model-free off-policy RL algorithm for image-based continuous control. DrQ-v2 is an extension of the original DrQ method [18], which adopts an actor–critic method with data augmentation to learn directly from image pixels. In comparison to DrQ, DrQ-v2 significantly improves sample efficiency, running approximately 3.5 times faster. This substantial improvement is achieved through several algorithmic modifications: (i) DrQ-v2 switches the underlying RL algorithm from SAC [14] to DDPG [15], (ii) this change enables the straightforward integration of multi-step return, contributing to enhanced performance, (iii) bilinear interpolation is introduced to the random shift augmentation technique, further benefiting performance, (iv) an exploration schedule is implemented to enhance exploration during training, (v) DrQ-v2 selects improved hyper-parameters, including a larger capacity for the replay buffer, leading to better overall results. Notably, DrQ-v2 stands out as the first model-free RL agent to successfully tackle complex 3D humanoid locomotion tasks from the DMC suite while learning directly from image pixels. The critic is optimized by the following loss:

$$\begin{aligned}&\mathcal{L}_Q(\phi_k,\xi,\mathcal{D})\\ &= \mathbb{E}_{\tau\sim\mathcal{D}}\left[(Q_{\phi_k}(\boldsymbol{h}_t,\boldsymbol{a}_t) - y_t)^2\right] \quad \forall k\in\{1,2\},\end{aligned} \quad (2)$$

where TD target, y_t, is defined as follows:

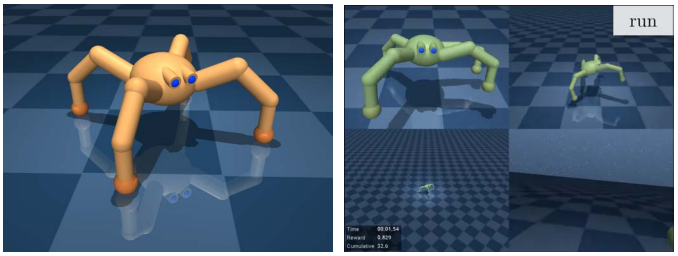
$$y_t = \sum_{i=0}^{n-1}\gamma^i r_{t+i} + \gamma^n \min_{k=1,2} Q_{\bar{\phi}_k}(\boldsymbol{h}_{t+n},\boldsymbol{a}_{t+n}),$$

where $\boldsymbol{h}_t = (f_\xi(\texttt{aug}(\boldsymbol{o}_t))$, $\boldsymbol{h}_{t+n} = f_\xi(\texttt{aug}(\boldsymbol{o}_{t+n}))$, and $\boldsymbol{a}_{t+n} = \pi_\theta(\boldsymbol{h}_{t+n}) + \epsilon$. Here, ϵ represents exploration noise sampled from $\text{clip}(\mathcal{N}(0,\sigma^2), -c, c)$, which is similar to TD3 [19], and $\bar{\phi}_1$ and $\bar{\phi}_2$ represent the exponential moving averages of the weights for the Q target networks.

The actor is optimized by the following loss:

$$\mathcal{L}_\pi(\theta,\mathcal{D}) = -\mathbb{E}_{\boldsymbol{o}_t\sim\mathcal{D}}\left[\min_{k=1,2} Q_{\phi_k}(\boldsymbol{h}_t,\boldsymbol{a}_t)\right], \quad (3)$$

where $\boldsymbol{h}_t = f_\xi(\texttt{aug}(\boldsymbol{o}_t)$ and $\boldsymbol{a}_t = \pi_\theta(\boldsymbol{h}_t) + \epsilon$.



(a) (b)

Fig. 2: Two 3D locomotion tasks from the DMC suite: (a) ‘quadruped-walk’ and (b) ‘quadruped-run.’

IV. Experiments

A. Experimental Setup

1) Environment Setup: We consider an environment for 3D continuous control that allows for image pixels as observation. To be specific, we consider learning directly from image pixels. In this setup, we do not use the 56 coordinate states provided by the environment, but instead take as input an image stack of three consecutive RGB images of size 84×84. The action space consists of a total of 12 elements, where there are 4 legs with 3 actuators on each leg.

2) Task Setup: We consider the DMC suite [13], a widely used benchmark for continuous control problems. We consider two representative 3D locomotion tasks (‘quadruped-walk’ and ‘quadruped-run’) among a variety of tasks from the DMC suite. Figure 2 shows the two 3D locomotion tasks, ‘quadruped-walk’ and ‘quadruped-run.’

3) Reward: We consider each episode with 1,000 environment steps for the two 3D locomotion tasks, where a per-step reward is within [0, 1].

4) Training: We use PyTorch as a deep learning tool. Within the experiments, we use a workstation with an Intel i9 CPU and Nvidia Quadro RTX 8000 GPU. During training, we ran five different random seeds to provide a reliable comparative evaluation; that is, the experimental results are averaged over five different seeds.

B. Empirical Evaluation

We conduct an empirical evaluation for the three recent visual RL algorithms, including CURL, SAC+AE, and DrQ-v2, with respect to sample efficiency and task performance with respect to the two 3D locomotion tasks (‘quadruped-walk’ and ‘quadruped-run’) from the DMC suite. As shown in Figure 3, our experimental results show that CURL based on contrastive learning and SAC+AE based on an auto-encoder struggle to solve the two 3D locomotion tasks whereas DrQ-v2 based on data augmentation performs well on the tasks. In the case of the task ‘quadruped-walk,’ both CURL and SAC+AE are not able to solve the tasks even within 3M environment steps. These results indicate that using data augmentation, rather than using contrastive learning and an auto-encoder, plays an important role in improving sample efficiency and task performance in image-based training. Therefore, only

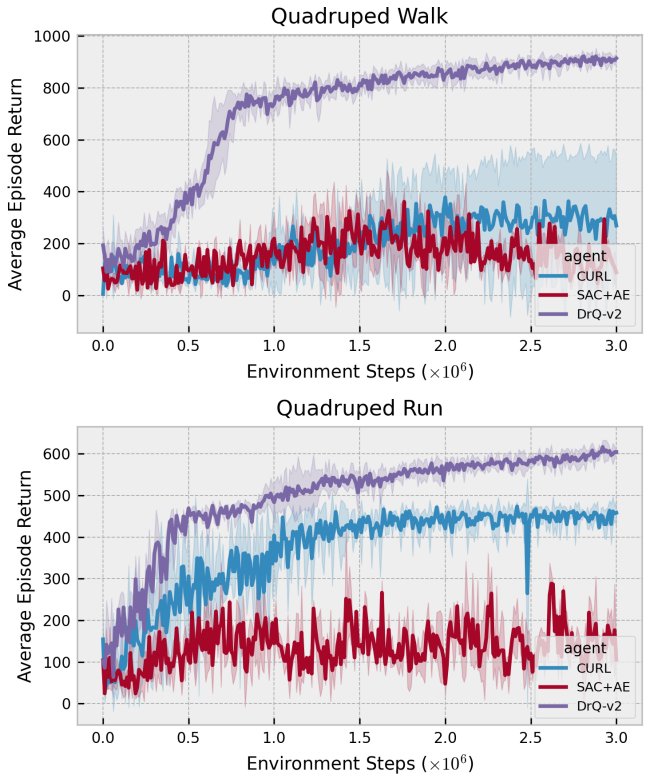


Fig. 3: Experimental results for the three visual RL algorithms (CURL, SAC+AE, and DrQ-v2) on the two 3D locomotion tasks (‘quadruped-walk’ and ‘quadruped-run’) from the DMC suite.

using an unsupervised manner through contrastive learning or an auto-encoder without applying data augmentations may not be a good choice for efficiently learning a policy in 3D continuous control.

V. Conclusion

We have conducted an empirical investigation of visual RL, which can be trained end-to-end directly from image pixels, to address 3D continuous control. Specifically, we have evaluated three recent visual RL algorithms, including CURL, SAC+AE, and DrQ-v2, with respect to sample efficiency and task performance on two 3D locomotion tasks (‘quadruped-walk’ and ‘quadruped-run’) from the DMC suite. Our experimental results show that CURL based on contrastive learning and SAC+AE based on an auto-encoder struggle to solve the two 3D locomotion tasks whereas DrQ-v2 based on data augmentation performs well on the tasks; that is, DrQ-v2 outperforms CURL and SAC+AE on the two 3D locomotion tasks with respect to sample efficiency and task performance. These results indicate that using data augmentation, rather than using contrastive learning or an auto-encoder, plays an important role in improving sample efficiency and task performance in image-based training. In future work, we hope to investigate incorporating vision with additional data (for

example, proprioceptive and tactile feedback) rather than using visual feedback only.

ACKNOWLEDGMENT

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. [23ZR1100, A Study of Hyper-Connected Thinking Internet Technology by autonomous connecting, controlling and evolving ways].

REFERENCES

[1] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *nature* 518.7540 (2015): 529-533.

[2] Levine, Sergey, et al. "End-to-end training of deep visuomotor policies." *The Journal of Machine Learning Research* 17.1 (2016): 1334-1373.

[3] Ibarz, Julian, et al. "How to train your robot with deep reinforcement learning: lessons we have learned." *The International Journal of Robotics Research* 40.4-5 (2021): 698-721.

[4] Canovas, Bruce, Amaury Nègre, and Michèle Rombaut. "Onboard dynamic RGB-D simultaneous localization and mapping for mobile robot navigation." *ETRI Journal* 43.4 (2021): 617-629.

[5] Jung, Sunggoo, et al. "Collision-free local planner for unknown subterranean navigation." *ETRI Journal* 43.4 (2021): 580-593.

[6] Hong, Seonghun, et al. "Special issue on recent advancements in simultaneous localization and mapping (SLAM) and its applications." *ETRI Journal* 43.4 (2021): 577-579.

[7] Seo, Seonghun, and Hoon Jung. "A robust collision prediction and detection method based on neural network for autonomous delivery robots." *ETRI Journal* 45.2 (2023): 329-337.

[8] Yu, W. and Song, S., "Design and experimentation of remote driving system for robotic speed sprayer operating in orchard environment," *ETRI Journal* 45 (2023), 479-491.

[9] Laskin, Michael, Aravind Srinivas, and Pieter Abbeel. "Curl: Contrastive unsupervised representations for reinforcement learning." *International Conference on Machine Learning*. PMLR, 2020.

[10] Yarats, Denis, et al. "Improving sample efficiency in model-free reinforcement learning from images." Proceedings of *the AAAI Conference on Artificial Intelligence*. Vol. 35. No. 12. 2021.

[11] Yarats, Denis, et al. "Mastering visual continuous control: Improved data-augmented reinforcement learning." *arXiv preprint* arXiv:2107.09645 (2021).

[12] Hafner, Danijar, et al. "Mastering atari with discrete world models." *arXiv preprint* arXiv:2010.02193 (2020).

[13] Tassa, Yuval, et al. "Deepmind control suite." *arXiv preprint* arXiv:1801.00690 (2018).

[14] T. Haarnoja, et al., "Soft actor-critic algorithms and applications," *arXiv preprint* arXiv:1812.05905, 2018.

[15] T. P. Lillicrap, et al., "Continuous control with deep reinforcement learning," *arXiv preprint* arXiv:1509.02971, 2015.

[16] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. "Deterministic policy gradient algorithms." *International Conference on Machine Learning*, pages 387-395. PMLR, 2014.

[17] Higgins, Irina, et al. "beta-vae: Learning basic visual concepts with a constrained variational framework." *International Conference on Learning Representations*. 2016.

[18] Kostrikov, Ilya, Denis Yarats, and Rob Fergus. "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels." *arXiv preprint* arXiv:2004.13649 (2020).

[19] Scott Fujimoto, Herke Hoof, and David Meger. "Addressing function approximation error in actor-critic methods." *International Conference on Machine Learning*, pages 1587-1596. PMLR, 2018.